Mantis Documentation

Release 0.2.1

Siemens

July 21, 2015

Contents

The MANTIS (Model-based Analysis of Threat Intelligence Sources) Framework consists of several Django Apps that, in combination, support the management of cyber threat intelligence expressed in standards such as STIX, CybOX, OpenIOC, IODEF (RFC 5070), etc.

Important resources:

- Access to the Mantis source code for installation:
  - Either via `git clone` from the Mantis Github Repository (recommended):

```
git clone https://github.com/siemens/django-mantis.git
```

  - Or via download as `zip` package from https://github.com/siemens/django-mantis/archive/master.zip
- There is a mailing list for dicussions, questions, etc.:
  - Subscribe to the mailing list by sending a mail to `Mantis-ti-discussion-join@lists.trusted-introducer.`
  - The archives of the mailing list are available via Nabble.

  Many thanks to the TF-CSIRT Trusted Introducer for their support in hosting the list!
- All issues regarding Mantis and its components are tracked on the Mantis Issue Tracker.
- Documentation:

CHAPTER 1

MANTIS Architecture

The MANTIS (Model-based Analysis of Threat Intelligence Sources) Framework consists of several Django Apps that, in combination, support the management of cyber threat intelligence expressed in standards such as STIX, CybOX, OpenIOC, IODEF (RFC 5070), etc.

The heavy lifting is done in the following Django Apps:

- django-dingos
- django-mantis-core
- django-mantis-stix-importer
- django-mantis-openioc-importer
- django-mantis-iodef-importer
- django-mantis-taxii (under development)

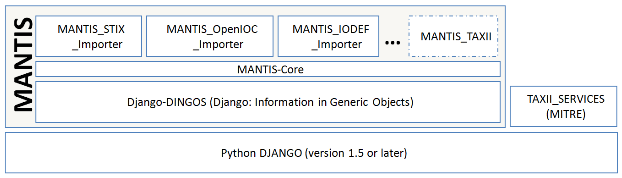


Fig. 1.1: MANTIS architecture

CHAPTER 2

Screenshots

To get an idea of what MANTIS currently provides, take a look at the following screenshots.

Contents

2.1 Login

Django's standard login screen, rendered with the Grappelli skin that is used by Mantis. You can customize Django to do authentication differently (see the Django documentation on customizing authentication.)

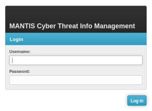


Fig. 2.1: The login screen

2.2 Menus

In its default configuration, MANTIS currently presents three menus:

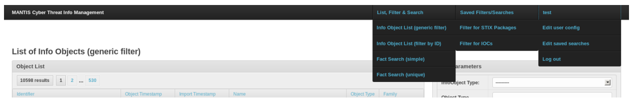
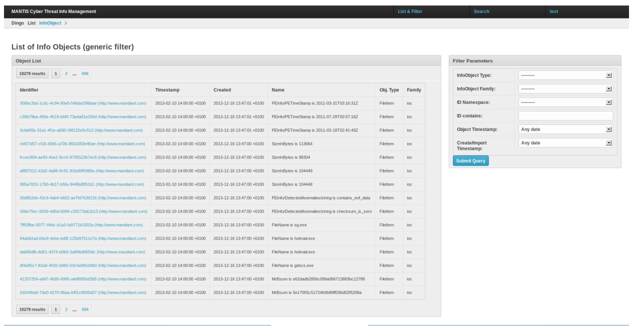


Fig. 2.2: The menus presented to the user by MANTIS

- A menu over which the existing search/filter views are accessible
- A menu over which saved searches are accessible
- A menu for viewing/editing user-specific information

2.3 Viewing imported information objects

The screenshot below shows the overview of imported information objects right after import of MITRE's conversion of the MITRE STIX conversion of APT-1 report. We imported the top-level STIX package and the Appendix G with full indicators of compromise (i.e., Mandiant OpenIOC is embedded into the STIX XML). The count shows a quite large number of objects, and we obviously need a way to find our way around. So in the next step, we filter the list a bit.



List of Info Objects (generic filter)

Identifier	Timestamp	Created	Name	Obj. Type	Family
300bc2bd-1cdc-4c94-90e0-54bba1f9bbae (http://www.mandiant.com)	2013-02-10 14:00:00 +0100	2013-12-16 13:47:01 +0100	PEInfo/PETimeStamp is 2011-03-31T03:16:31Z	FileItem	ioc
c39b79ba-460e-4619-bf49-73a4a81e256d (http://www.mandiant.com)	2013-02-10 14:00:00 +0100	2013-12-16 13:47:01 +0100	PEInfo/PETimeStamp is 2011-07-29T00:57:16Z	FileItem	ioc
3cfaf45b-31a1-4f1e-a690-09f132e5c612 (http://www.mandiant.com)	2013-02-10 14:00:00 +0100	2013-12-16 13:47:01 +0100	PEInfo/PETimeStamp is 2011-03-18T02:41:49Z	FileItem	ioc
cbf27d57-cf18-40b5-a706-8501083e46ae (http://www.mandiant.com)	2013-02-10 14:00:00 +0100	2013-12-16 13:47:00 +0100	SizeInBytes is 113664	FileItem	ioc
fccec804-ae93-4ea1-9cc6-8795523b7ec6 (http://www.mandiant.com)	2013-02-10 14:00:00 +0100	2013-12-16 13:47:00 +0100	SizeInBytes is 98304	FileItem	ioc
af887012-42d2-4a98-9c91-91fa99f5986a (http://www.mandiant.com)	2013-02-10 14:00:00 +0100	2013-12-16 13:47:00 +0100	SizeInBytes is 104449	FileItem	ioc
995a7833-1780-4b17-b5fa-944f6d8f51b1 (http://www.mandiant.com)	2013-02-10 14:00:00 +0100	2013-12-16 13:47:00 +0100	SizeInBytes is 104448	FileItem	ioc
30d852eb-43c9-4ab4-b602-ae7fd7636216 (http://www.mandiant.com)	2013-02-10 14:00:00 +0100	2013-12-16 13:47:00 +0100	PEInfo/DetectedAnomalies/string is contains_eof_data	FileItem	ioc
266e75ec-5639-4d5d-b094-c59173a61b13 (http://www.mandiant.com)	2013-02-10 14:00:00 +0100	2013-12-16 13:47:00 +0100	PEInfo/DetectedAnomalies/string is checksum_is_zero	FileItem	ioc
7ff03fbe-0077-44dc-b1a3-fa9771b3302a (http://www.mandiant.com)	2013-02-10 14:00:00 +0100	2013-12-16 13:47:00 +0100	FileName is sg.exe	FileItem	ioc
94ab92ad-b5e9-4ebe-bd9f-125b97511e7a (http://www.mandiant.com)	2013-02-10 14:00:00 +0100	2013-12-16 13:47:00 +0100	FileName is hotmail.exe	FileItem	ioc
da666dfb-6d51-4374-b0b0-3a896d06f3dc (http://www.mandiant.com)	2013-02-10 14:00:00 +0100	2013-12-16 13:47:00 +0100	FileName is hotmail.exe	FileItem	ioc
df3e85c7-82a9-4032-b860-03c5e891d3b0 (http://www.mandiant.com)	2013-02-10 14:00:00 +0100	2013-12-16 13:47:00 +0100	FileName is gdocs.exe	FileItem	ioc
41207254-a9d7-4b95-9080-a4d8905d2fd5 (http://www.mandiant.com)	2013-02-10 14:00:00 +0100	2013-12-16 13:47:00 +0100	Md5sum is e62dadb2856c099a066713883bc12788	FileItem	ioc
2d244ba9-73e0-4270-96aa-64f1c8935d27 (http://www.mandiant.com)	2013-02-10 14:00:00 +0100	2013-12-16 13:47:00 +0100	Md5sum is 5e17055c51724b0b89ff036d02f5208a	FileItem	ioc

InfoObject Type:	----------
InfoObject Family:	----------
ID Namespace:	----------
ID contains:	
Object Timestamp:	Any date
Create/Import Timestamp:	Any date

Fig. 2.3: The list of information objects (standard URL: `/mantis/View/InfoObject`)

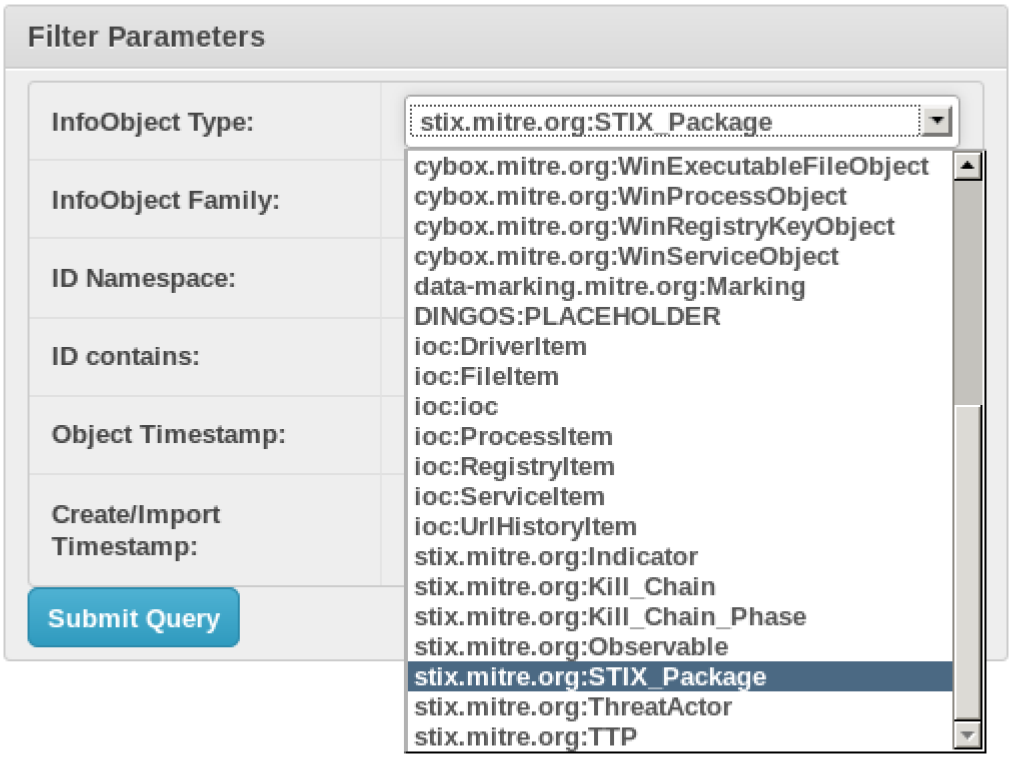
2.4 Filtering

The filter box on the page showing the information object list allows filtering with respect to several commonly used criteria. Here, we filter by information object type, and chose the `STIX_Package`.

Filtering results for `STIX_Packages` yields two results: the package that represents the top-level of the APT-1 report and the package that represents appendix G.

2.5 Viewing an info object

Clicking on the STIX package for the top-level of the APT-1 report shows MANTIS's representation of the info object:

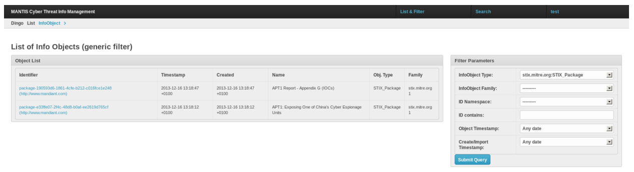


Filter Parameters

InfoObject Type:	stix.mitre.org:STIX_Package
InfoObject Family:	
ID Namespace:	
ID contains:	
Object Timestamp:	
Create/Import Timestamp:	

cybox.mitre.org:WinExecutableFileObject
cybox.mitre.org:WinProcessObject
cybox.mitre.org:WinRegistryKeyObject
cybox.mitre.org:WinServiceObject
data-marking.mitre.org:Marking
DINGOS:PLACEHOLDER
ioc:DriverItem
ioc:FileItem
ioc:ioc
ioc:ProcessItem
ioc:RegistryItem
ioc:ServiceItem
ioc:UrlHistoryItem
stix.mitre.org:Indicator
stix.mitre.org:Kill_Chain
stix.mitre.org:Kill_Chain_Phase
stix.mitre.org:Observable
stix.mitre.org:STIX_Package
stix.mitre.org:ThreatActor
stix.mitre.org:TTP
Submit Query

Fig. 2.4: Filtering with respect to information object types



List of Info Objects (generic filter)

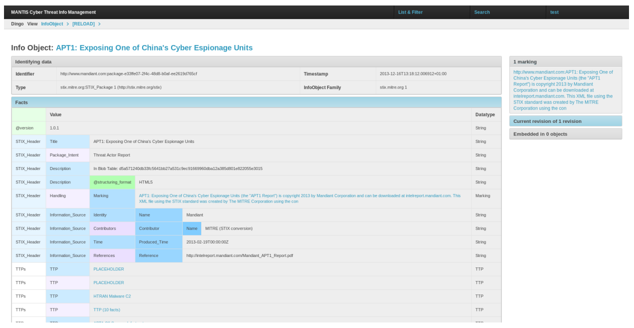
Object List

Identifier	Timestamp	Created	Name	Obj. Type	Family
package-190593d6-1861-4cfe-b212-c016fce1e248 (http://www.mandiant.com)	2013-12-16 13:18:47 +0100	2013-12-16 13:18:47 +0100	APT1 Report - Appendix G (IOCs)	STIX_Package	stix.mitre.org 1
package-e33ffe07-2f4c-48d8-b0af-ee2619d765cf (http://www.mandiant.com)	2013-12-16 13:18:12 +0100	2013-12-16 13:18:12 +0100	APT1: Exposing One of China's Cyber Espionage Units	STIX_Package	stix.mitre.org 1

Filter Parameters

InfoObject Type:	stix.mitre.org:STIX_Package
InfoObject Family:	---------
ID Namespace:	---------
ID contains:	
Object Timestamp:	Any date
Create/Import Timestamp:	Any date

Fig. 2.5: Result of filtering for `STIX_Packages`



MANTIS Cyber Threat Info Management
List & Filter
Search
test
Dingo View InfoObject > [RELOAD] >

Info Object: APT1: Exposing One of China's Cyber Espionage Units

Identifying data

Identifier	http://www.mandiant.com:package-e33ffe07-2f4c-48d8-b0af-ee2619d765cf	Timestamp	2013-12-16T13:18:12.006912+01:00
Type	stix.mitre.org:STIX_Package 1 (http://stix.mitre.org/stix)	InfoObject Family	stix.mitre.org 1

Facts

	Value					Datatype
@version	1.0.1					String
STIX_Header	Title	APT1: Exposing One of China's Cyber Espionage Units				String
STIX_Header	Package_Intent	Threat Actor Report				String
STIX_Header	Description	In Blob Table: d5a571240db33fc5641bb27a531c9ec91669960dba12a385d801e822055e3015				String
STIX_Header	Description	@structuring_format	HTML5			String
STIX_Header	Handling	Marking	APT1: Exposing One of China's Cyber Espionage Units (the "APT1 Report") is copyright 2013 by Mandiant Corporation and can be downloaded at intelreport.mandiant.com. This XML file using the STIX standard was created by The MITRE Corporation using the con			Marking
STIX_Header	Information_Source	Identity	Name	Mandiant		String
STIX_Header	Information_Source	Contributors	Contributor	Name	MITRE (STIX conversion)	String
STIX_Header	Information_Source	Time	Produced_Time	2013-02-19T00:00:00Z		String
STIX_Header	Information_Source	References	Reference	http://intelreport.mandiant.com/Mandiant_APT1_Report.pdf		String
TTPs	TTP	PLACEHOLDER				TTP
TTPs	TTP	PLACEHOLDER				TTP
TTPs	TTP	HTRAN Malware C2				TTP
TTPs	TTP	TTP (10 facts)				TTP

1 marking

http://www.mandiant.com:APT1: Exposing One of China's Cyber Espionage Units (the "APT1 Report") is copyright 2013 by Mandiant Corporation and can be downloaded at intelreport.mandiant.com. This XML file using the STIX standard was created by The MITRE Corporation using the con

Current revision of 1 revision

Embedded in 0 objects

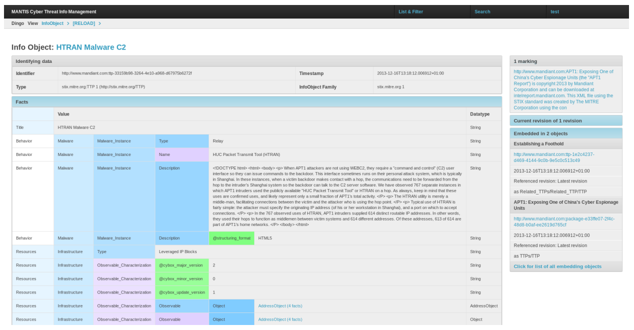
Fig. 2.6: View of STIX package presenting top-level of APT 1 report

- At the top, we have identifiying information.
- The bulk of the display in the center concerns the facts contained in the object (the color coding shows the structuring of the facts – it takes a bit of getting used to ... but this is just a view after all: you can create a view that suits you better.)

  The fact values that appear in blue are actually links to other info objects that have been extracted from the STIX package. You see two objects called `PLACEHOLDER`: as it turns out, the STIX package references these two objects without actually defining them. Would they be imported at a later point of time (identified by identifier and namespace of the identifier), the placeholders would be overwritten.
- The view also shows the marking that has been extracted and associated with this info object and all other info objects extracted from the STIX package.
- Curently, there is a single revision of the object in the system. If there were more revisions, these would be shown (as well as whether the revision you are looking at is the most recent revision).
- This information object is not embedded in another info object; if it were, information about these objects would be displayed.

2.6 Viewing another info object

Clicking on the value of the third fact with fact term `TTPs\TTP`, we see the facts contained in this info object ... and now there is also information about info objects in which this info object is embedded.



MANTIS Cyber Threat Info Management
List & Filter
Search
test
Dingo View InfoObject > [RELOAD] >

Info Object: HTRAN Malware C2

Identifying data

Identifier	http://www.mandiant.com:ttp-33159b98-3264-4e10-a968-d67975b6272f	Timestamp	2013-12-16T13:18:12.006912+01:00
Type	stix.mitre.org:TTP 1 (http://stix.mitre.org/TTP)	InfoObject Family	stix.mitre.org 1

Facts

	Value					Datatype
Title	HTRAN Malware C2					String
Behavior	Malware	Malware_Instance	Type	Relay		String
Behavior	Malware	Malware_Instance	Name	HUC Packet Transmit Tool (HTRAN)		String
Behavior	Malware	Malware_Instance	Description	<!DOCTYPE html> <html> <body> <p> When APT1 attackers are not using WEBC2, they require a "command and control" (C2) user interface so they can issue commands to the backdoor. This interface sometimes runs on their personal attack system, which is typically in Shanghai. In these instances, when a victim backdoor makes contact with a hop, the communications need to be forwarded from the hop to the intruder's Shanghai system so the backdoor can talk to the C2 server software. We have observed 767 separate instances in which APT1 intruders used the publicly available "HUC Packet Transmit Tool" or HTRAN on a hop. As always, keep in mind that these uses are confirmed uses, and likely represent only a small fraction of APT1's total activity. </P> <p> The HTRAN utility is merely a middle-man, facilitating connections between the victim and the attacker who is using the hop point. </P> <p> Typical use of HTRAN is fairly simple: the attacker must specify the originating IP address (of his or her workstation in Shanghai), and a port on which to accept connections. </P> <p> In the 767 observed uses of HTRAN, APT1 intruders supplied 614 distinct routable IP addresses. In other words, they used their hops to function as middlemen between victim systems and 614 different addresses. Of these addresses, 613 of 614 are part of APT1's home networks. </P> </body> </html>		String
Behavior	Malware	Malware_Instance	Description	@structuring_format	HTML5	String
Resources	Infrastructure	Type	Leveraged IP Blocks			String
Resources	Infrastructure	Observable_Characterization	@cybox_major_version	2		String
Resources	Infrastructure	Observable_Characterization	@cybox_minor_version	0		String
Resources	Infrastructure	Observable_Characterization	@cybox_update_version	1		String
Resources	Infrastructure	Observable_Characterization	Observable	Object	AddressObject (4 facts)	AddressObject
Resources	Infrastructure	Observable_Characterization	Observable	Object	AddressObject (4 facts)	Object

1 marking

http://www.mandiant.com:APT1: Exposing One of China's Cyber Espionage Units (the "APT1 Report") is copyright 2013 by Mandiant Corporation and can be downloaded at intelreport.mandiant.com. This XML file using the STIX standard was created by The MITRE Corporation using the con

Current revision of 1 revision

Embedded in 2 objects

Establishing a Foothold

http://www.mandiant.com:ttp-1e2c4237-d469-4144-9c0b-9e5c0c513c49

2013-12-16T13:18:12.006912+01:00

Referenced revision: Latest revision

as Related_TTPs/Related_TTP/TTP

APT1: Exposing One of China's Cyber Espionage Units

http://www.mandiant.com:package-e33ffe07-2f4c-48d8-b0af-ee2619d765cf

2013-12-16T13:18:12.006912+01:00

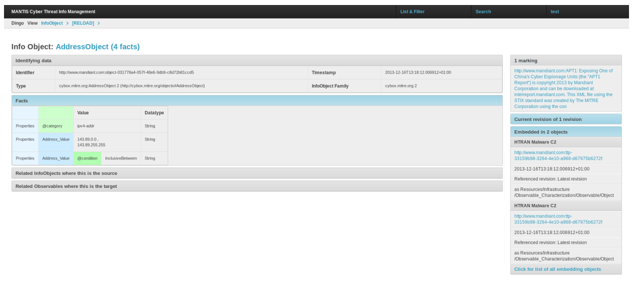
Referenced revision: Latest revision

as TTPs/TTP

Click for list of all embedding objects

Fig. 2.7: Viewing a TTP object. Standard URL for viewing is `mantis/View/InfoObject/<object-nr>`

Clicking once more, this time into an address object (here, the pre-defined naming schema did not work and produced the name `AddressObject (4 facts)` – but you can configure additional naming schemas), we view another info object:



MANTIS Cyber Threat Info Management
List & Filter
Search
test
Dingo View InfoObject > [RELOAD] >

Info Object: AddressObject (4 facts)

Identifying data

Identifier	http://www.mandiant.com:object-031778a4-057f-48e6-9db9-c8d72b81ccd5	Timestamp	2013-12-16T13:18:12.006912+01:00
Type	cybox.mitre.org:AddressObject 2 (http://cybox.mitre.org/objects#AddressObject)	InfoObject Family	cybox.mitre.org 2

Facts

		Value		Datatype
Properties	@category	ipv4-addr		String
Properties	Address_Value	143.89.0.0 , 143.89.255.255		String
Properties	Address_Value	@condition	InclusiveBetween	String

Related InfoObjects where this is the source

Related Observables where this is the target

1 marking

http://www.mandiant.com:APT1: Exposing One of China's Cyber Espionage Units (the "APT1 Report") is copyright 2013 by Mandiant Corporation and can be downloaded at intelreport.mandiant.com. This XML file using the STIX standard was created by The MITRE Corporation using the con

Current revision of 1 revision

Embedded in 2 objects

HTRAN Malware C2

http://www.mandiant.com:ttp-33159b98-3264-4e10-a968-d67975b6272f

2013-12-16T13:18:12.006912+01:00

Referenced revision: Latest revision

as Resources/Infrastructure/Observable_Characterization/Observable/Object

HTRAN Malware C2

http://www.mandiant.com:ttp-33159b98-3264-4e10-a968-d67975b6272f

2013-12-16T13:18:12.006912+01:00

Referenced revision: Latest revision

as Resources/Infrastructure/Observable_Characterization/Observable/Object

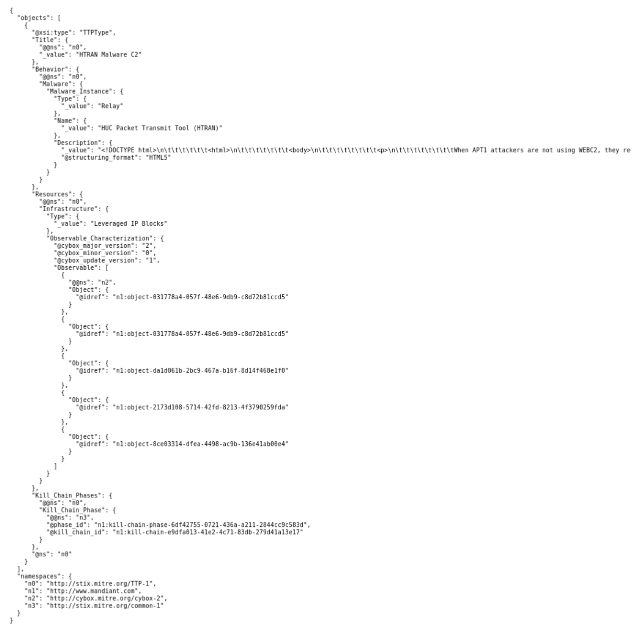
Click for list of all embedding objects

Fig. 2.8: Viewing an address object

Again, we have information about which objects this particular object is embedded in: we get two results, and two times the same object, because it has been referenced two times (once by mistake, it seems.)

2.7 Viewing the JSON representation

Mantis stores objects internally as lists of facts (refer to the DINGOS model description to learn more about the internal data model), but can also produce a JSON representation of each object.



```
{
  "objects": [
    {
      "@xsi:type": "TTPType",
      "Title": {
        "@@ns": "n0",
        "_value": "HTRAN Malware C2"
      },
      "Behavior": {
        "@@ns": "n0",
        "Malware": {
          "Malware_Instance": {
            "Type": {
              "_value": "Relay"
            },
            "Name": {
              "_value": "HUC Packet Transmit Tool (HTRAN)"
            },
            "Description": {
              "_value": "<!DOCTYPE html>\n\t\t\t\t\t\t<html>\n\t\t\t\t\t\t\t<body>\n\t\t\t\t\t\t\t\t<p>\n\t\t\t\t\t\t\t\tWhen APT1 attackers are not using WEBC2, they re
              "@structuring_format": "HTML5"
            }
          }
        }
      },
      "Resources": {
        "@@ns": "n0",
        "Infrastructure": {
          "Type": {
            "_value": "Leveraged IP Blocks"
          },
          "Observable_Characterization": {
            "@cybox_major_version": "2",
            "@cybox_minor_version": "0",
            "@cybox_update_version": "1",
            "Observable": [
              {
                "@@ns": "n2",
                "Object": {
                  "@idref": "n1:object-031778a4-057f-48e6-9db9-c8d72b81ccd5"
                }
              },
              {
                "Object": {
                  "@idref": "n1:object-031778a4-057f-48e6-9db9-c8d72b81ccd5"
                }
              },
              {
                "Object": {
                  "@idref": "n1:object-da1d061b-2bc9-467a-b16f-8d14f468e1f0"
                }
              },
              {
                "Object": {
                  "@idref": "n1:object-2173d108-5714-42fd-8213-4f3790259fda"
                }
              },
              {
                "Object": {
                  "@idref": "n1:object-8ce03314-dfea-4498-ac9b-136e41ab00e4"
                }
              }
            ]
          }
        }
      },
      "Kill_Chain_Phases": {
        "@@ns": "n0",
        "Kill_Chain_Phase": {
          "@@ns": "n3",
          "@phase_id": "n1:kill-chain-phase-6df42755-0721-436a-a211-2844cc9c583d",
          "@kill_chain_id": "n1:kill-chain-e9dfa013-41e2-4c71-83db-279d41a13e17"
        }
      },
      "@ns": "n0"
    }
  ],
  "namespaces": {
    "n0": "http://stix.mitre.org/TTP-1",
    "n1": "http://www.mandiant.com",
    "n2": "http://cybox.mitre.org/cybox-2",
    "n3": "http://stix.mitre.org/common-1"
  }
}
```

Fig. 2.9: JSON representation of a STIX TTP object. Standard url is `mantis/View/InfoObject/<object-nr>/json`

Unfortunately, the JSON representation has still a slight problem: in the last few lines, the identifiers for `@phase_id` and `@kill_chain_id` would have to be treated akin to the "normal" references using `idref`.

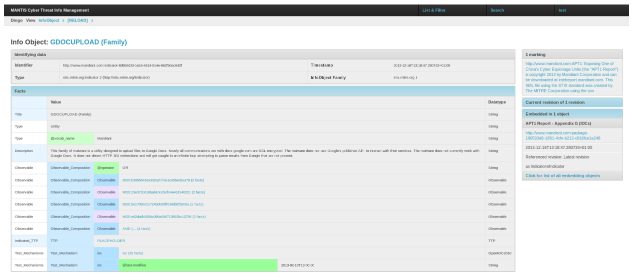
2.8 Dealing with embeddings of different standards

STIX is very flexible and allows the embedding of other standards, such as Mandiant's OpenIOC. For example, the MITRE STIX conversion of APT-1 report contains one version of the "Appendix G", that contains embedded OpenIOC indicators. The Mantis STIX importer recognizes such occurrences and hands off to the Mantis OpenIOC importer.

Clicking on the embedded `ioc` object (here, the naming went wrong, it should display the value of the `short_description` element in the IOC) in line `Test_Mechanisms/Test_Mechanism/ioc` yields a view of the imported OpenIOC info object.

2.9 Searching and viewing results

We also can search for facts:



MANTIS Cyber Threat Info Management | List & Filter | Search | test

Dingo View InfoObject > [RELOAD] >

Info Object: GDOCUPLOAD (Family)

Identifying data

Identifier	http://www.mandiant.com:indicator-8d88dd33-1e16-4814-814e-662fb0ac842f	**Timestamp**	2013-12-16T13:18:47.280733+01:00
Type	stix.mitre.org:Indicator 2 (http://stix.mitre.org/Indicator)	**InfoObject Family**	stix.mitre.org 1

Facts

	Value				Datatype
Title	GDOCUPLOAD (Family)				String
Type	Utility				String
Type	@vocab_name	Mandiant			String
Description	This family of malware is a utility designed to upload files to Google Docs. Nearly all communications are with docs.google.com are SSL encrypted. The malware does not use Google's published API to interact with their services. The malware does not currently work with Google Docs. It does not detect HTTP 302 redirections and will get caught in an infinite loop attempting to parse results from Google that are not present.				String
Observable	Observable_Composition	@operator	OR		String
Observable	Observable_Composition	Observable	MD5:b305b543da332a2fcf6e1ce55ed2ea79 (2 facts)		Observable
Observable	Observable_Composition	Observable	MD5:23e371b816bab10cd9cfc4a46154022c (2 facts)		Observable
Observable	Observable_Composition	Observable	MD5:5e17055c51724b0b89ff036d02f5208a (2 facts)		Observable
Observable	Observable_Composition	Observable	MD5:e62dadb2856c099a066713883bc12788 (2 facts)		Observable
Observable	Observable_Composition	Observable	AND (... (4 facts)		Observable
Indicated_TTP	TTP	PLACEHOLDER			TTP
Test_Mechanisms	Test_Mechanism	ioc	ioc (35 facts)		OpenIOC2010
Test_Mechanisms	Test_Mechanism	ioc	@last-modified	2013-02-10T13:00:00	String

1 marking

http://www.mandiant.com:APT1: Exposing One of China's Cyber Espionage Units (the "APT1 Report") is copyright 2013 by Mandiant Corporation and can be downloaded at intelreport.mandiant.com. This XML file using the STIX standard was created by The MITRE Corporation using the con

Current revision of 1 revision

Embedded in 1 object

APT1 Report - Appendix G (IOCs)

http://www.mandiant.com:package-190593d6-1861-4cfe-b212-c016fce1e248

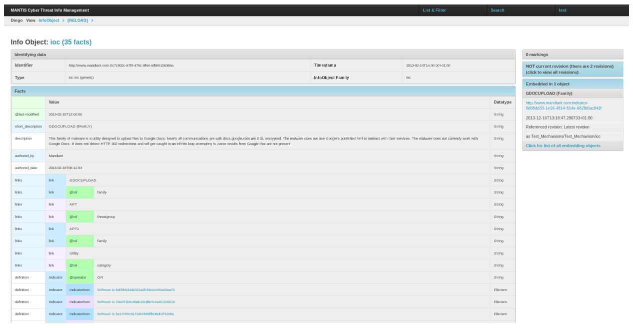
2013-12-16T13:18:47.280733+01:00

Referenced revision: Latest revision

as Indicators/Indicator

Click for list of all embedding objects

Fig. 2.10: STIX indicator with embedded OpenIOC indicator (fact with fact term `Test_Mechanisms/Test_Mechanism/ioc`).



MANTIS Cyber Threat Info Management | List & Filter | Search | test

Dingo View InfoObject > [RELOAD] >

Info Object: ioc (35 facts)

Identifying data

Identifier	http://www.mandiant.com:0c7c902c-67f8-479c-9f44-4d985106365a	**Timestamp**	2013-02-10T14:00:00+01:00
Type	ioc:ioc (generic)	**InfoObject Family**	ioc

Facts

	Value			Datatype
@last-modified	2013-02-10T13:00:00			String
short_description	GDOCUPLOAD (FAMILY)			String
description	This family of malware is a utility designed to upload files to Google Docs. Nearly all communications are with docs.google.com are SSL encrypted. The malware does not use Google's published API to interact with their services. The malware does not currently work with Google Docs. It does not detect HTTP 302 redirections and will get caught in an infinite loop attempting to parse results from Google that are not present.			String
authored_by	Mandiant			String
authored_date	2013-02-10T06:11:53			String
links	link	GDOCUPLOAD		String
links	link	@rel	family	String
links	link	APT		String
links	link	@rel	threatgroup	String
links	link	APT1		String
links	link	@rel	family	String
links	link	Utility		String
links	link	@rel	category	String
definition	Indicator	@operator	OR	String
definition	Indicator	IndicatorItem	Md5sum is b305b543da332a2fcf6e1ce55ed2ea79	FileItem
definition	Indicator	IndicatorItem	Md5sum is 23e371b816bab10cd9cfc4a46154022c	FileItem
definition	Indicator	IndicatorItem	Md5sum is 5e17055c51724b0b89ff036d02f5208a	FileItem

0 markings

NOT current revision (there are 2 revisions) (click to view all revisions)

Embedded in 1 object

GDOCUPLOAD (Family)

http://www.mandiant.com:indicator-8d88dd33-1e16-4814-814e-662fb0ac842f

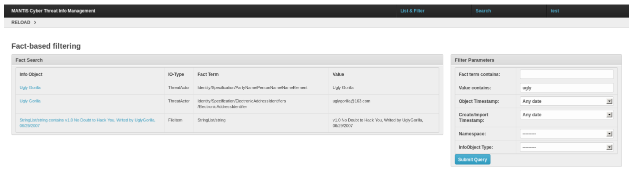
2013-12-16T13:18:47.280733+01:00

Referenced revision: Latest revision

as Test_Mechanisms/Test_Mechanism/ioc

Click for list of all embedding objects

Fig. 2.11: An OpenIOC indicator



MANTIS Cyber Threat Info Management | List & Filter | Search | test

RELOAD >

Fact-based filtering

Fact Search

Info Object	IO-Type	Fact Term	Value
Ugly Gorilla	ThreatActor	Identity/Specification/PartyName/PersonName/NameElement	Ugly Gorilla
Ugly Gorilla	ThreatActor	Identity/Specification/ElectronicAddressIdentifiers /ElectronicAddressIdentifier	uglygorilla@163.com
StringList/string contains v1.0 No Doubt to Hack You, Writed by UglyGorilla, 06/29/2007	FileItem	StringList/string	v1.0 No Doubt to Hack You, Writed by UglyGorilla, 06/29/2007

Filter Parameters

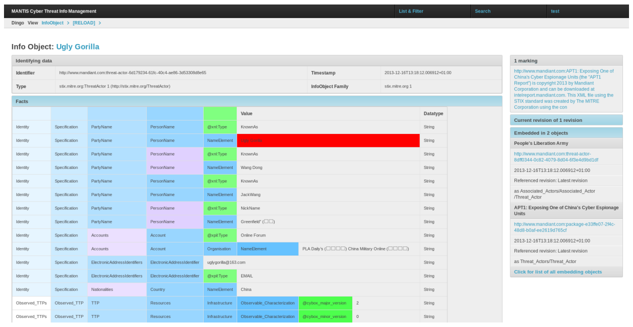
Fact term contains:	
Value contains:	ugly
Object Timestamp:	Any date
Create/Import Timestamp:	Any date
Namespace:	---------
InfoObject Type:	---------

Submit Query

Fig. 2.12: Searching for values

The search page allows us to search for values, e.g. the word `ugly`. This yields several results. The display shows the info objects in which the value occurs, the info object type of these objects, and the fact term under which the value occurs.

Clicking on one of the objects shows the object and marks in red the occurrence of the searched term.


MANTIS Cyber Threat Info Management
List & Filter
Search
test
Dingo View InfoObject > [RELOAD] >

Info Object: Ugly Gorilla

Identifying data

Identifier	http://www.mandiant.com:threat-actor-6d179234-61fc-40c4-ae86-3d53308d8e65	Timestamp	2013-12-16T13:18:12.006912+01:00
Type	stix.mitre.org:ThreatActor 1 (http://stix.mitre.org/ThreatActor)	InfoObject Family	stix.mitre.org 1

Facts

					Value			Datatype
Identity	Specification	PartyName	PersonName	@xnl:Type	KnownAs			String
Identity	Specification	PartyName	PersonName	NameElement	Ugly Gorilla			String
Identity	Specification	PartyName	PersonName	@xnl:Type	KnownAs			String
Identity	Specification	PartyName	PersonName	NameElement	Wang Dong			String
Identity	Specification	PartyName	PersonName	@xnl:Type	KnownAs			String
Identity	Specification	PartyName	PersonName	NameElement	JackWang			String
Identity	Specification	PartyName	PersonName	@xnl:Type	NickName			String
Identity	Specification	PartyName	PersonName	NameElement	Greenfield" (□□)			String
Identity	Specification	Accounts	Account	@xpil:Type	Online Forum			String
Identity	Specification	Accounts	Account	Organisation	NameElement	PLA Daily's (□□□□) China Military Online (□□□□)		String
Identity	Specification	ElectronicAddressIdentifiers	ElectronicAddressIdentifier	uglygorilla@163.com				String
Identity	Specification	ElectronicAddressIdentifiers	ElectronicAddressIdentifier	@xpil:Type	EMAIL			String
Identity	Specification	Nationalities	Country	NameElement	China			String
Observed_TTPs	Observed_TTP	TTP	Resources	Infrastructure	Observable_Characterization	@cybox_major_version	2	String
Observed_TTPs	Observed_TTP	TTP	Resources	Infrastructure	Observable_Characterization	@cybox_minor_version	0	String

1 marking

http://www.mandiant.com:APT1: Exposing One of China's Cyber Espionage Units (the "APT1 Report") is copyright 2013 by Mandiant Corporation and can be downloaded at intelreport.mandiant.com. This XML file using the STIX standard was created by The MITRE Corporation using the con

Current revision of 1 revision

Embedded in 2 objects

People's Liberation Army

http://www.mandiant.com:threat-actor-8dff0344-0c82-4079-8d04-6f3e4d9bd1df

2013-12-16T13:18:12.006912+01:00

Referenced revision: Latest revision

as Associated_Actors/Associated_Actor /Threat_Actor

APT1: Exposing One of China's Cyber Espionage Units

http://www.mandiant.com:package-e33ffe07-2f4c-48d8-b0af-ee2619d765cf

2013-12-16T13:18:12.006912+01:00

Referenced revision: Latest revision

as Threat_Actors/Threat_Actor

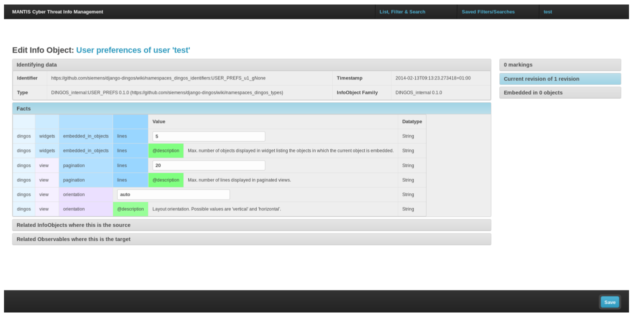
Click for list of all embedding objects

Fig. 2.13: Viewing a search result

2.10 Editing user-specific data

Currently, each user can edit his user configurations and saved searches.

2.10.1 Edit user configurations



Edit Info Object: User preferences of user 'test'

Identifying data

Identifier	https://github.com/siemens/django-dingos/wiki/namespaces_dingos_identifiers:USER_PREFS_u1_gNone	Timestamp	2014-02-13T09:13:23.273418+01:00
Type	DINGOS_internal:USER_PREFS 0.1.0 (https://github.com/siemens/django-dingos/wiki/namespaces_dingos_types)	InfoObject Family	DINGOS_internal 0.1.0

Facts

				Value		Datatype
dingos	widgets	embedded_in_objects	lines	5		String
dingos	widgets	embedded_in_objects	lines	@description	Max. number of objects displayed in widget listing the objects in which the current object is embedded.	String
dingos	view	pagination	lines	20		String
dingos	view	pagination	lines	@description	Max. number of lines displayed in paginated views.	String
dingos	view	orientation	auto			String
dingos	view	orientation	@description	Layout orientation. Possible values are 'vertical' and 'horizontal'.		String

Related InfoObjects where this is the source

Related Observables where this is the target

0 markings

Current revision of 1 revision

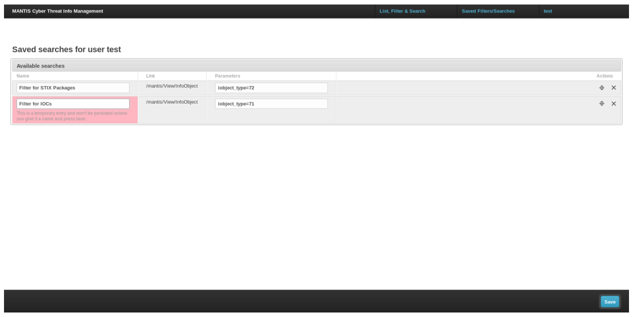
Embedded in 0 objects

Fig. 2.14: The view for editing the user configurations

Currently, there is only a minimum of user configurations available – these will be extended in future releases of MANTIS. Also, the framework for managing user configurations is very flexible and can be used for own development (see the relevant documentation of DINGOS.)

2.10.2 Edit user configurations

After pressing the 'Save Search' button on the filter view, users are presented with a view that allows them to add the new search and edit the exiting ones; the view is also available via the user-specific menu in the top right of the screen.



Saved searches for user test

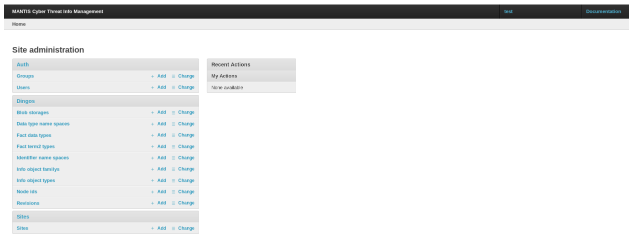
Available searches

Name	Link	Parameters	Actions
Filter for STIX Packages	/mantis/View/InfoObject	iobject_type=72	
Filter for IOCs This is a temporary entry and won't be persisted unless you give it a name and press save.	/mantis/View/InfoObject	iobject_type=71	

Fig. 2.15: The view for editing saved searches

2.11 A look at the admin interface

Django features a very powerful admin interface. We us it to view and manage enumerables such as info object types, fact data types, etc.



Site administration

Auth		
Groups	+ Add	Change
Users	+ Add	Change
Dingos		
Blob storages	+ Add	Change
Data type name spaces	+ Add	Change
Fact data types	+ Add	Change
Fact term2 types	+ Add	Change
Identifier name spaces	+ Add	Change
Info object familys	+ Add	Change
Info object types	+ Add	Change
Node ids	+ Add	Change
Revisions	+ Add	Change
Sites		
Sites	+ Add	Change

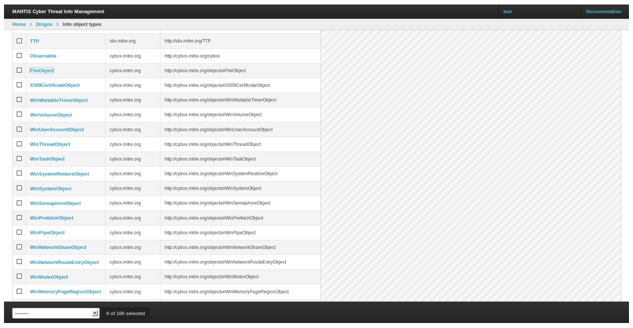
Recent Actions

My Actions

None available

Fig. 2.16: The Django admin interface with overview of DINGOS's models

For example, here the list of info object types in the system.

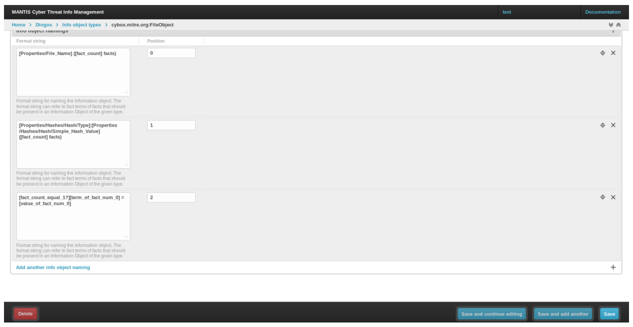

MANTIS Cyber Threat Info Management
test
Documentation
Home › Dingos › Info object types

☐	TTP	stix.mitre.org	http://stix.mitre.org/TTP
☐	Observable	cybox.mitre.org	http://cybox.mitre.org/cybox
☐	FileObject	cybox.mitre.org	http://cybox.mitre.org/objects#FileObject
☐	X509CertificateObject	cybox.mitre.org	http://cybox.mitre.org/objects#X509CertificateObject
☐	WinWaitableTimerObject	cybox.mitre.org	http://cybox.mitre.org/objects#WinWaitableTimerObject
☐	WinVolumeObject	cybox.mitre.org	http://cybox.mitre.org/objects#WinVolumeObject
☐	WinUserAccountObject	cybox.mitre.org	http://cybox.mitre.org/objects#WinUserAccountObject
☐	WinThreadObject	cybox.mitre.org	http://cybox.mitre.org/objects#WinThreadObject
☐	WinTaskObject	cybox.mitre.org	http://cybox.mitre.org/objects#WinTaskObject
☐	WinSystemRestoreObject	cybox.mitre.org	http://cybox.mitre.org/objects#WinSystemRestoreObject
☐	WinSystemObject	cybox.mitre.org	http://cybox.mitre.org/objects#WinSystemObject
☐	WinSemaphoreObject	cybox.mitre.org	http://cybox.mitre.org/objects#WinSemaphoreObject
☐	WinPrefetchObject	cybox.mitre.org	http://cybox.mitre.org/objects#WinPrefetchObject
☐	WinPipeObject	cybox.mitre.org	http://cybox.mitre.org/objects#WinPipeObject
☐	WinNetworkShareObject	cybox.mitre.org	http://cybox.mitre.org/objects#WinNetworkShareObject
☐	WinNetworkRouteEntryObject	cybox.mitre.org	http://cybox.mitre.org/objects#WinNetworkRouteEntryObject
☐	WinMutexObject	cybox.mitre.org	http://cybox.mitre.org/objects#WinMutexObject
☐	WinMemoryPageRegionObject	cybox.mitre.org	http://cybox.mitre.org/objects#WinMemoryPageRegionObject

----
0 of 100 selected

Fig. 2.17: Admin overview of the info object types

Access to the info object types via the admin interface is especially relevant, because naming schemas that govern how objects are named are defined per info object type.


MANTIS Cyber Threat Info Management
test
Documentation
Home > Dingos > Info object types > cybox.mitre.org:FileObject
Add another info object naming
Delete
Save and continue editing
Save and add another
Save

Format string	Position
[Properties/File_Name] ([fact_count] facts)	0
[Properties/Hashes/Hash/Type]:[Properties/Hashes/Hash/Simple_Hash_Value] ([fact_count] facts)	1
[fact_count_equal_1?][term_of_fact_num_0] = [value_of_fact_num_0]	2

Format string for naming the information object. The format string can refer to fact terms of facts that should be present in an Information Object of the given type.

Fig. 2.18: Configuration of naming schemas for file objects

CHAPTER 3

What MANTIS is and isn't

MANTIS

- *isn't* a finished tool or even project: we like to think that it provides a solid basis on which cyber-threat intelligence management can be built up upon, but if you expect something that out of the box covers all aspects of cyber-threat intelligence management, MANTIS isn't for you.
- (currently) *isn't* a tool fit for importing *huge* datasets. It can import fairly large XML documents such as the MITRE STIX conversion of the APT-1 report, but this takes a while (expect 20-30 minutes or so.) So do not expect to be able to throw, e.g., dozens and dozens of MAEC files with sizes of several 100MBs into the system: the generic importer is not fit for such sizes.

  This situation may change at some point of time with more stream-lined importers, but MANTIS is really not intended to deal with very big data the way log management solutions such as Splunk et al. are.

What MANTIS is:

- MANTIS provides an example implementation of a framework for managing cyber threat intelligence expressed in standards such as STIX, CybOX, IODEF, etc. The aims of providing such an example implementation are:
  * To aide discussions about emerging standards such as STIX, CybOX et al. with respect to questions regarding tooling: how would a certain aspect be implemented, how do changes affect an implementation? Such discussions become much easier and have a better basis if they can be lead in the context of example tooling that is known to the community.
  * To lower the entrance barrier for organizations and teams (esp. CERT teams) in using emerging standards for cyber-threat intelligence management and exchange.
  * To provide a platform on the basis of which research and community-driven development in the area of cyber-threat intelligence management can occur.
- Even though MANTIS is in no way a complete system, it already does cover a first use case: MANTIS provides an information repository into which cyber threat intelligence received in STIX/CybOX, OpenIOC and IODEF can be imported in a meaningful way that allows browsing, filtering and searching for information. Thus, MANTIS can be used as information base for keeping all the information you receive and information you generate yourself that is expressed in one of the currently supported standards. Because the importer is highly configurable, importers for other structured data should not be too difficult to write (and will hopefully be shared with the community ...).

CHAPTER 4

History

4.1 0.2.1 (2014-03-06)

- Changed dependencies for Mantis components
  - Mantis now requires DINGOS in version 0.2.1. The differences to 0.2.0 are as follows:
    - Bugfixes
    - *CRITICAL* Remediation of painfully slow import for systems with lot's of imported data

      An illformed query led to extremely slow import of new data in systems that already have lot's of data inside. This bug has been fixed.
    - Problem in link to InfoObjects in which a certain fact can be found on Unique Search Page fixed

      The link was faulty in that it carried a '&page=...' parameter that needed to be removed.
    - Long repetition of '_' in a string lead to HTML display spilling over, because '_' was not regarded as place to insert a possible line break. This has been changed.
    - New/Modified views
    - View for listing *all* InfoObjects, also those used internally by DINGOS for bookkeeping (e.g., user preferences). The view is restricted to Django-superusers.
    - New/Modified command-line commands
    - In 'dingos_manage_user_settings', added the ability to overwrite settings for 'ALL' users.

4.2 0.2.0 (2014-02-26)

- Changed dependencies for Mantis components
  - Mantis now requires DINGOS in version 0.2.0. The differences to 0.1.0 are as follows:
    - New base functionality
    - Added framework for managing user-specific data (user configurations, saved searches, etc.) and querying user-specific data in templates and views.
    - Added tracking of namespace information per component of a fact term
    - New/Modified views
    - Modifications to all views

- Added possibility to switch between horizontal and vertical layout ... or have automatic adjustment of the layout depending on screen width.
- Modifications to filter views
- Modified date-picker in filters to enable addition of timespans without changing saved searches or messing up order of timespans
- Added several further filter criteria in InfoObject filter
- Added view with basic and still rather restricted editing capabilities for InfoObjects – currently only used for editing user preferences or edits by the superuser
- Added view to edit user configuration
- Added view to edit saved searches
- Added per-column ordering to list views
- Added new filter/search that shows unique Facts rather than all InfoObjects containing a certain fact.
- New/added capabilities for writing views
- Added framework for ordering list views
- Added per-user configuration for:
- layout (horizontal vs. vertical)
- number of rows to show in list views
- number of rows to show in widget displaying objects in which a displayed object is embedded
- Bug fixes / Improvements
- Generation of filter views became unbearably slow when many (> 40,000) InfoObjects are in the system. This was, because of a badly built query within the dynamically built filter form. This has been fixed.
- Further development of JSON export (still needs work to make the to_dict function of InfoObjects generic and configurable such as the from_dict function)
- Fixed bug in generation of InfoObjects: when a placeholder for a given ID already existed, it was not reliably found.
- New/Modified command-line commands
- Import command now fails gracefully if import of a file throws an exception: it continues with import of the next file.
- Added command line arguments to basic import command:
- ability to add IDs of marking objects to be added to imported objects
- ability to automatically move imported XML files to other folder after import
- Added command to reset user-settings and saved searches for a given user.
- Added command to re-calculate object names.

  This is useful to run right after an import, recalculating the names of 'Observable' InfoObjects created in the past few minutes. Thus, the problem that those Observables that are to be named after the (single) object they contain do not carry a proper name (because at creation time of the Observable, the Object usually does not exist, yet) can be fixed.

* Mantis now requires the Mantis-Core in version 0.2.0. The differences to 0.1.0 are as follows:

        - Added corresponding abstract model classes for models introduced in DINGOS 0.2.0.
    - Mantis now requires the STIX/CybOX Importer in version 0.2.0. The differences to 0.1.0 are as follows:
        - Added ability to generate identifier for top-level element (usually a STIX_Package) if an identifier for that element is missing: if a default namespace has been defined, then an identifier is generated by taking the MD5-hash of the xml file.
        - Markings present in STIX_Package are read out and attached to all InfoObjects generated from the STIX_Package.

          Note: Mantis does currently not interpret the XPATH expression that specifies the scope of the marking (which is not much of an issue, since it seems that the feature to restrict the scope of a marking is not much used at the moment).
        - Timestamp present in *STIX_Header/Information_Source/Time/Produced_Time* is read.
        - Added a command-line argument to add a default-timestamp to the STIX import command.
        - Bug fixes:
        - Attributes other than *id* and *idref* that contained a namespace were not handled correctly. The handler function *attr_with_namespace_handler* fixes this.
        - In *0.1.0*, the *xsi:type* attribute was not recorded, because in most cases, its information is used for determining the data type of elements and InfoObjects. But there are cases, e.g., in Markings, where this is not the case. For these cases, the *xsi:type* attribute is kept in the InfoObject.
        - Family revision info was not recorded; this has been fixed.
    - Mantis now requires the OpenIOC Importer in version 0.2.0. The differences to 0.1.0 are as follows:
        - Fixed bug in import of timestamp.

4.3 0.1.0 (2013-12-19)

- Initial release

CHAPTER 5

Installation

Contents

5.1 Manual installation

ATTENTION: Please do not use the master branch for fresh installs; use the development branch and follow the installation instructions outlined in the development branch

The installation instructions below have been tested on an out-of-the-box installation of Ubuntu Desktop 12.04 LTS (the Desktop rather than the Server version has been used, since the majority of installs are likely to be for testing and developing, where having a full working environment and X-server installed comes in handy.) If you are using a different *nix flavor, you have to find the corresponding installation packages used with `apt-get` below – the installation steps carried out with `pip`, however, will be exactly the same.

Attention: If you are setting up a virtual machine, make sure to give it at least 3GB of memory if you want to import really large XML structures such as MITRE's STIX conversion of the Mandiant APT-1 report (http://stix.mitre.org/downloads/APT1-STIX.zip) – importing large files currently takes a lot of memory – there seems to be a memory leak which we still have to track down.

1. Make sure that you have the required dependencies on OS level for building the XML-related packages. For example, on an Ubuntu system, execute the following commands:

```
$ apt-get install libxml2 libxml2-dev python-dev libxslt1-dev
```

Also, while you are at it, install git, if you do not have it already:

```
$ apt-get install git
```

If you are behind a proxy, you can configure a proxy for `apt-get` by putting a file `95proxy` into `/etc/apt/apt.conf.d` that has the following contents:

```
Acquire::http::proxy "<proxy_url>";
Acquire::ftp::proxy "<proxy_url>";
Acquire::https::proxy "<proxy_url>";
```

2. It is recommended to use a virtual python environment.

   – Make sure that `virtualenv` and `pip` are installed:

```
$ apt-get install python-virtualenv python-pip
```

   – Create a virtual environment:

```
$ virtualenv <path_for_storing_environments>/mantis
$ source <path_for_storing_environments>/mantis/bin/activate
```

   Now the virtual environment is activated – you should see a changed prompt that is prefixed with `(mantis)`

3. Install the `libxml2-python` bindings

   Unfortunately, the process of getting `libxml2-python` installed using `pip` varies from OS to OS, because there is no proper library package available.

   – For Ubuntu 12.04 do the following:

   Use pip to install directly from an ftp source:

```
(mantis)$ pip install ftp://xmlsoft.org/libxml2/python/libxml2-python-2.6.21.tar.gz
```

   If you are behind a proxy, you can either provide `pip` with the proxy information with the commandline argument `--proxy <proxy_url>` or use the `http_proxy` environment variable – do not do both, because this confuses `pip`.

   If the download does not work via pip, download the file with your browser, and install from file with `pip install <filename>`.

   – For Ubuntu 13.10, do the following:

     * Downloadand unpack the `libxml2` sources:

```
(mantis)$ wget http://xmlsoft.org/sources/libxml2-2.9.1.tar.gz"
(mantis)$ tar -zxvf libxml2-2.9.1.tar.gz
```

     If you are behind a proxy, append `-e use_proxy=yes -e http_proxy=<proxy_url>` after the URL when calling `wget`.

     * Install via `pip`:

```
(mantis)$ pip install libxml2-2.9.1/python
```

     If you are behind a proxy, you can either provide `pip` with the proxy information with the commandline argument `--proxy <proxy_url>` or use the `http_proxy` environment variable – do not do both, because this confuses `pip`.

4. Go to a location where you want to have the Django Mantis files and check out the git repository:

```
(mantis)$ git clone https://github.com/siemens/django-mantis.git
```

   If you are behind a proxy, you can configure a proxy for `git` via the following:

```
(mantis)$ git config --global http.proxy <proxy_url>
```

5. Change into the `django-mantis` directory and do:

```
(mantis)$ pip install -r requirements/local.txt
(mantis)$ pip install django-simple-menu>=1.0.6
```

(For some reason, `django-simple-menu` cannot be installed before Django itself has not been installed completely).

6. Your are now all set for running MANTIS on top of an SQLite database. If that is what you want to do, have a look at QUICKSTART.
7. For running MANTIS on top of Postgresql (which is recommended), you need to install and prepare Postgresql:
   - Install it:

```
$ apt-get install postgresql
$ apt-get install postgresql-server-dev-9.1
```

   - Install the Python module for working with postgresql:

```
(mantis)$ pip install psycopg2
```

   - In `/etc/postgresql/9.1/main/postgresql.conf` set `ssl = False`
   - (Re)start the server:

```
/etc/init.d/postgresql start
```

   - Create password for `postgresql`: as root user, do:

```
passwd postgres
```

   - Give the postgresql user a database password; As user `postgres` do:

```
su postgres
psql
\password postgres;
```

   - Prepare database:
     * As user postgresql do:

```
createuser -P mantis;
```

     and do the following:

       · give it password `mantis`
       · do not make it super user
       · allow it to create databases (required for running python unit tests). If you forgot about this step here, you can later run (`ALTER USER mantis CREATEDB;`) on the database prompt to achieve the same.
       · do not allow it to create new roles

   - In database, do:

```
CREATE DATABASE django OWNER mantis ENCODING 'UTF-8';
```

  - In `/etc/postgresql/9.1/main/pg_hba.conf` enter after the line for the postgres user:

```
# TYPE  DATABASE        USER            ADDRESS                 METHOD

local [tab] django [tab] mantis [tab][tab]  md5
```

8. Continue with the QUICKSTART.

5.2 (Semi-)automated installation with Vagrant

Vagrant allows automated provisioning of virtual machines with preconfigured packages, configuration settings, etc.

(Development of Vagrant deployment scripts for Mantis is an ongoing effort. Scripts and documentation will be published as they become ready.)

QUICKSTART

In the `django-mantis` folder, do the following:

- For easy demo usage with SQLite, do:

```
(mantis)$ bash quickstart.sh
```

  (Note that this uses a SQLite database file located in the `/tmp` directory: any imports you do in Mantis will therefore not survive a system restart. You can move the location of the SQLite database by modifying the line reading `'/tmp/django-mantis_test.db'` in `mantis/settings/local.py`.)

- For usage with exisiting and configured postgresql database, do:

```
(mantis)$  bash quickstart_psql
```

The script will ask, whether at this stage, you want to create an administrative user for Django. Answer with *yes* and provide user name, email address and password.

In detail, the bash script will do the following:

1. Run the Django `syncdb` command, which
   (a) creates tables for the models of all applications that are *not* using the Django South application for database migrations.
   (b) asks you for user name, email address and password of an administrative Django user (you will need this username and password later to log on)
2. Carry out (initial) database migrations for all MANTIS components using the South migrations that are part of the components' distribution (in subdirectory `migrations`)
3. Configure default naming schemata for the exisiting importer modules of MANTIS via calling the command `mantis_<format>_set_naming` for each such module
4. Carry out the Django `collect_static` command, which copies over the static files for all applications to the `static` folder configured in the settings of MANTIS
5. Show you (via the `less` command) this file and (after you quit `less`), print the file to the console
6. Start the testing web server running MANTIS via Django's `runserver` command on port 8000.

Then try out the following:

- Download: http://stix.mitre.org/downloads/APT1-STIX.zip and extract the files
- For the files Mandiant_APT1_Report.xml and Appendix_G_IOCs_Full.xml do the following:
  - ∗ If you are using sqllite:

```
python manage.py mantis_stix_import --settings=mantis.settings.local  --trace\
    --marking_json=quickstart_examples/markings/minimal_marking.json\
    --marking_pfill=source "Mandiant APT 1 Report"\
    <file_path>
```

  * If you are using postgresql:

```
python manage.py mantis_stix_import --settings=mantis.settings.local_psql  --trace\
    --marking_json=quickstart_examples/markings/minimal_marking.json\
    --marking_pfill=source "Mandiant APT 1 Report"\
    <file_path>
```

Start with Mandiant_APT1_Report.xml: that goes relatively fast; Appendix_G_IOCs_Full.xml will take about 20 minutes or so to import.

ATTENTION: The import of large files takes quite a bit of memory (probably there is a memory leak somewhere, which will be ironed out in a future release). Be sure to give the system/virtual machine you are running the import of `Appendix_G_IOCs_Full.xml` on a fair amount of memory (4 GB definitely works).

– Start the server (if the quickstart-script has not started it already for you) with

  * If you are using sqllite:

```
python manage.py runserver 8000 --traceback --settings=mantis.settings.local
```

  * If you are using postgresql:

```
python manage.py runserver 8000 --traceback --settings=mantis.settings.local_psql
```

– Browse to:

```
127.0.0.1:8000/mantis/View/InfoObject
```

and start looking around:

- Select a filter for `stix.mitre.org:STIX_Package` in the filter box in the top-right corner.
- This will show you all `STIX_Package` objects that are in the system (two, if you imported both `Mandiant_APT1_Report.xml` and `Appendix_G_IOC_Full.xml`).
- Click on one of the two objects and start exploring (have a look at the screenshots in the documentation for a quick guide through the application.)

You can also have a look at the Django admin interface at:

```
127.0.0.1:8000/admin
```

CHAPTER 7

MANTIS developers’ guide

Contents:

7.1 Before starting to develop

7.1.1 Read up on techniques and styles used in MANTIS

MANTIS profitted a lot from the advice provided in Two Scoops of Django.

Unless you are an absolute Django expert (and maybe even then), please read Daniel Greenfield’s and Audrey Roy’s excellent Two Scoops of Django. Even though it provides best practices for Django 1.5, most of its advice is also valid for Django 1.6, and likely to be very relevant for quite a few minor revisions to come.

7.1.2 Understand how django-dingos works

The heart of MANTIS is the django-dingos Django application. Most aspects of modifying/adding to MANTIS will require a sound understanding of how *django-dingos* works. Please refer to the Django DINGOS developers’ guide

7.1.3 Find the right place to modify/add to

Writing your own Django application

If you are adding completely new functionality to Mantis, the best way may very well be to create a new Django application.

Keep django-dingos generic

Although DINGOS is likely to be used mainly in the context of the Django MANTIS Cyber Threat Intelligence Management application, DINGOS should stay a /generic/ application for managing structured information. So whenever you find yourself adding/modifying stuff in DINGOS that is specific to cyber threat intelligence management, the STIX, CybOX standards, etc., **DINGOS is the wrong place to modify/add to**. The same goes for customizations that are particular to your instance of running MANTIS.

Please consider the following places for development instead:

- If you want to add Python code that is particular to cyber threat management, consider adding this in django-mantis-core
- If you want to add Python code that is particular to a certain standard, consider adding it to the respective importer module, e.g., django-mantis-stix-importer or similar
- If you want to make modifications to a DINGOS template that is required for your local instance of MANTIS (or whatever framework is using DINGOS), the right way is probably to override one of the DINGOS base templates. Have a look at how django-mantis overrides the `templates/dingos/grappelli/base.html` template; see also the Django documentation on overriding templates.
- If you want to change the url paths of DINGOS views, do this in the `url.py` of your instance rather than `dingos/url.py`.

7.2 Setting up a development environment

1. Refer to Contributing (section "Getting Started") for information of how to (1) either fork a repository, clone it, and install it for development purposes, or (2) set up the directory structure for your own Django app that will contribute to the Mantis framework.
2. Chose a development environment of your liking. Here is how you can setup PyCharm Professional Edition in support of development for Django: * Start up PyCharm and enter your license information. * Before opening a project/folder, go to `Configure -> Settings` and adjust the following:
   - Use the search box in the settings dialog to find the place where you can configure the proxy settings:
   - Configure the python environment under "Project Interpreter" -> "Python Interpreters" Click on the "+", then on "Local..." Select `<path_to_your_environment>/bin/python`, and click "Ok"
   - Click on "Ok" to close the settings window.
   - Open the project folder: select "Open Directory" and choose your source directories
   - Before being able to run the django-mantis project, you have to adjust the "Run/Debug Confgurations" (wait for the indexer to finish...)
     * In the menubar, click on "Run" -> "Edit Configurations"
     * Select the "django-mantis" in the displayed tree on the left
     * In the right pane, add the following to the "Additional options:" `--settings=mantis.settings.local_psql` or `--settings=mantis.settings.local`
   - You should now be able to run the django server by clicking the play button.

7.3 MANTIS Application Layout

Contents

7.3.1 Overview of the directory layout

The layout of the DINGOS Django application is as follows:

```
.
-- mantis
|   -- apps
|   -- assets
|   -- blobs
|   -- menus.py
|   -- models.py
|   -- settings
|   |   -- base.py
|   |   -- local_psql.py
|   |   -- local.py
|   |   -- production.py
|   |   -- testing.py
|   -- static
|   -- templates
|   |   -- 404.html
|   |   -- 500.html
|   |   -- base.html
|   |   -- dingos
|   |   |   -- grappelli
|   |   |       -- base.html
|   |   -- mantis
|   |       -- grappelli
|   -- urls.py
|   -- wsgi.py
```

CHAPTER 8

Contributing

Contributions are welcome, and they are greatly appreciated! Every little bit helps, and credit will always be given.

You can contribute in many ways:

8.1 Types of Contributions

8.1.1 Report Bugs

MANTIS encompasses a number of components. For the following base components, please report issues at the central issue tracker for the whole Django MANTIS framework at https://github.com/siemens/django-mantis/issues :

- https://github.com/siemens/django-mantis
- https://github.com/siemens/django-dingos
- https://github.com/siemens/django-mantis-core
- https://github.com/siemens/django-mantis-openioc-importer
- https://github.com/siemens/django-mantis-stix-importer
- https://github.com/siemens/django-mantis-iodef-importer

If you are reporting a bug, please include:

- Your operating system name and version.
- Any details about your local setup that might be helpful in troubleshooting.
- Detailed steps to reproduce the bug.

8.1.2 Fix Bugs

Look through the GitHub issues for bugs. Anything tagged with “bug” is open to whoever wants to implement it.

8.1.3 Implement Features

Look through the GitHub issues for features. Anything tagged with "feature" is open to whoever wants to implement it.

8.1.4 Write Documentation

Djangos could always use more documentation, whether as part of the official Djangos docs, in docstrings, or even on the web in blog posts, articles, and such.

8.1.5 Submit Feedback

The best way to send feedback is to file an issue at https://github.com/siemens/django-mantis/issues.

If you are proposing a feature:

- Explain in detail how it would work.
- Keep the scope as narrow as possible, to make it easier to implement.
- Remember that this is a volunteer-driven project, and that contributions are welcome :)

8.2 Get Started!

In your contribution, you may want to either modify/add to existing code or create a new Django application that interacts with the existing applications that are part of the Mantis framework.

MANTIS profitted a lot from the advice provided in Two Scoops of Django. Unless you are an absolute Django expert (and maybe even then), please read Daniel Greenfield's and Audrey Roy's excellent Two Scoops of Django. Even though it provides best practices for Django 1.5, most of its advice is also valid for Django 1.6, and likely to be very relevant for quite a few minor revisions to come.

8.2.1 Modifying/adding to existing code

Here's how to set up a repository for local development.

1. Fork the relevant repository repo on GitHub.
2. Clone your fork locally:

```
$ git clone git@github.com:your_name_here/<repository>.git
```

3. Install your local copy into a virtualenv. Assuming you have virtualenvwrapper installed, this is how you set up your fork for local development:

```
$ mkvirtualenv <your_mantis_environment>
$ cd <repository_folder>
$ python setup.py develop
```

4. Create a branch for local development:

```
$ git checkout -b name-of-your-bugfix-or-feature
```

Now you can make your changes locally.

5. Commit your changes and push your branch to GitHub:

```
$ git add .
$ git commit -m "Your detailed description of your changes."
$ git push origin name-of-your-bugfix-or-feature
```

6. Submit a pull request through the GitHub website.

8.2.2 Writing your own Django application

Do yourself a favor and set up the directory structure of your Django application in the right way from the very start. The easiest way to do so is to use Daniel Greenfield's cookiecutter-djangopackage template (which uses Audrey Roy's excellent Cookiecutter for creating the directories): this layout has a very sensible directory structure with out-of-the-box configuration of `setup.py` for easy build, submission to PyPi, etc., as well as the start of a Sphinx documentation tree. Once you have the directory structure created, initialize a fresh git repository with it and get to work...

8.3 Pull Request Guidelines

Before you submit a pull request, check that it meets these guidelines:

1. The pull request should include tests.
2. If the pull request adds functionality, the docs should be updated. Put your new functionality into a function with a docstring, and add the feature to the list in README.rst.
3. The pull request should work for Python 2.7.